RAId: User Guide

Gelio Alves, Aleksey Ogurtsov, and Yi-Kuo Yu *

National Center for Biotechnology Information, National Library of Medicine, National Institutes of Health, Bethesda, MD 20894.

(Dated: November 30, 2012)

Contents

* to whom correspondence should be addressed: alves@ncbi.nlm.nih.gov,ogurtsov@ncbi.nlm.nih.gov,yyu@ncbi.nlm.nih.gov

A. RAId

RAId is a software tool designed to analyze tandem mass spectra. RAId can interpret tandem mass spectra and assign confidence to identified peptides by two different methods. In the first method RAId assigns statistical confidence to identified peptides using RAId_DbS statistics, and in the second method it uses RAId_aPS to assign statistics confidence to identified peptides. Both RAId_DbS[2] and RAId_aPS have several features which are briefly described below.

We will start first with RAId_DbS. RAId_DbS is a database search tool which uses the information contained in a tandem mass spectrum and searches a specified protein database for peptides that best explain the tandem spectrum. One of the many features of RAId_DbS is that for each identified peptide it reports an *E-value* based on a parametric distribution, whose form is derived theoretically[2] with parameters determined by the peptide scores collected, along with a model *P-value* informing the user how well the theoretical distribution agrees with the experimental score distribution [2].

RAId_DbS also has a unique database structure which incorporates information from different biological databases of already observed post-translational modifications (PTMs), proteotypic peptides, single amino acid polymorphisms (SAPs) and diseases associated with mutations [3]. The information used to construct RAId_DbS's annotated databases is obtained from various databases: NCBI's Gene Bank database, Swiss-Pro Protein Knowledgebase, Human Protein Reference Database (HPRD), ISB spectral library, NIST spectral library, and GPM spectral library. Annotated databases from various species that can be used by RAId_DbS are available for download from **ftp://ftp.nlm.nih.gov/pub/RAId/Software/RAId_DbS/**. RAId_DbS's unique database structure allows users to construct specialized databases based on the users own knowledge of PTMs, SAPs and diseases. The user constructed database can be searched separately or together with a specified organismal database provided by RAId_DbS.

We now briefly describe some features of RAId_aPS. RAId_aPS can be used as a database search tool and it uses the same database as RAId_DbS. RAId_aPS can also be used to generate the score distributions for all possible peptides [5] for a series of scoring functions. The scoring functions currently implemented in RAId_aPS are: RAId_DbS's scoring function [2], Hyperscore [7], XCorr [6] and Kscore [8]. The available scoring functions can be used to compute a combined database *P-value*, [4] which can increase peptide identification confidence and reduce the number false identifications.

Another feature of RAId_aPS is that it can be used to rescore and to compute accurate *P-values* and *E-values* for peptides in the output files of Mascot(.dat), SEQUEST(.out), X!Tandem (.xml), NIST (.msp), ISB (.msp), GPM (.mgf) and a list of peptides in a flat file. When rescoring the output files of Mascot(.dat), SEQUEST(.out) or X!Tandem (.xml) users do not need to specify the PTMs present in the individual files because the modifications will be automatically extracted from the input files by RAId_aPS and will be used to score the peptides present in the output files.

RAId_aPS scoring functions were implemented to resemble RAId_DbS score, SEQUEST XCorr, X!Tandem Hyperscore and X!Tandem Kscore plug-in. Because RAId_aPS can only incorporate scores transformable into additive parts, despite that many heuristics are included while different scoring functions were implemented in various programs, among these many heuristics only those that are additive were included in RAId_aPS. However, when users are analyzing the output files of Mascot(.dat), SEQUEST(.out) or X!Tandem (.xml) RAId_aPS can use the original score reported to compute an *E-value/P-value* for the original score base on an *E-value* calibration procedure described in a previous publication [1]. The *E-values* computed for the original scores will be combined with the *E-values* reported by the other scoring functions selected to produce an overall database combined *E-values*.

B. Software Package

Download Site:

ftp://ftp.ncbi.nlm.nih.gov/pub/RAId/Software/RAId_DbS/

Components:

Once properly installed the following files, executables and directory should be present in the installed directory.

RAId (executable file)
RaidDb (executable file)
UserDb.pl (perl file)
split_out.pl (perl file)
split_spectra.pl (perl file)
ReadMe.txt (text file)
RAId_Manual.pdf (pdf file)
RAId_parameters.txt (text file)
PARAMETER_FILES (directory)

The directory PARAMETER_FILES should contains the following files:

Score_Parameter_File (text file)
Parameter_File_Tag (text file)
RAId_DbS_PTM_file (text file)
RT_A100_Krokhin_AC_V78_N22_Y2006_P7785.txt (text file)
RT_A300_Krokhin_AC_V78_N22_Y2006_P7785.txt (text file)

Installation:

To install unzip and untar the file RAId.tar.gz. Once installed the above files should appear in the installed directory.

Downloadable Databases:

RAId performs searches in formatted protein databases. Special annotated and formatted protein databases for different organisms which are searchable by RAId are available for downloaded from the above ftp site.

C. Syntax

[−*ap*],[−*as*],[−*at*],
[−*b*],
[−*cev*],[−*cg*],
[−*daa*],[−*db*],[−*dnf*],[−*dsv*],[−*dt*],
[−*ed*],[−*evc*],[−*ex*],[−*exf*],[−*exo*],[−*ez*],[−*ect*],
[−*fp*],[−*fps*],[−*fpp*]
[−*gp*],
[−*ip*],[−*IL*]
[−*lb*],
[−*mc*],[−*mw*],
[−*ng*],[−*ns*],[−*nmcs*],
[−*of*],[−*op*]
[−*pf*],[−*pt*],[−*pfd*],
[−*QK*],
[−*rap*],[−*ras*], [−*rtf*],[−*rti*],[−*rto*],[−*rts*],
[−*ssh*],[−*spf*],[−*st*],
[−*tpp*],[−*tpf*],
[−*ub*],[−*ud*],[−*uid*],[−*up*],[−*urp*],
[−*v*],

Executing mode option:
[−*ex*], [−*b*]

Enzyme option:
[−*ez*], [−*ect*], [−*nmcs*]

Cysteine modification option:
[−*mc*]

Molecular weight options:
[−*cg*], [−*dt*], [−*mw*], [−*ng*], [−*pt*]

Amino acid residue (PTMs,SAPs) options:
[−*ap*], [−*as*], [−*daa*], [−*ns*], [−*rap*], [−*ras*], [−*up*], [−*urp*]

Proteotypic peptide options:
[−*at*]

Search options:
[−*lb*], [−*ub*]

Database options:
[−*db*], [−*fp*], [−*ud*]

Scoring options:
[−*cev*], [−*dsv*], [−*evc*], [−*sm*], [−*spf*]

Output and Input file options:
[−*dnf*], [−*exf*], [−*exo*], [−*ip*], [−*of*], [−*op*], [−*pf*]

Proportion of False Discovery options:
[−*pfd*], [−*IL*], [−*QK*], [−*tpf*]

D. Options

− ap

Number of annotated Post-Translation Modifications (PTMs) allowed per peptide.
Default value: $-ap$ 0.
Allowed parameter range $[0, 5]$.
$-ap$ 0 = does not search for annotated PTM.
$-ap$ 1 = 1 annotated PTM per peptide.
$-ap$ 2 = 2 annotated PTMs per peptide.

−as

Number of annotated Single Amino Acid Polymorphisms (SAPs) allowed per peptide.
Default value: $-as$ 0.
Allowed parameter $[0, 2]$
$-as$ 0 = does not search for annotated SAP.
$-as$ 1 = search for 1 annotated SAP per peptide.
$-as$ 2 = search for 2 annotated SAP per peptide.

−at

Checks if the peptide is proteotypic.
Default value: $-at$ 0.
$-at$ 0 = consider all peptides non-proteotypic.
$-at$ 1 = distinguish between proteotypic and non-proteotypic peptides.
$-at$ 2 = utilize only proteotypic peptides..

−b

When analysing files containing more than one single MS/MS spectrum the -b option is used to specify the name of a bash file which will be created with information for all the MS/MS spectra to be analysed. $-b$ file name (any).

−cev

This option is used with RAId_aPS.
It uses the original *E-value* reported in the output files from Mascot, SEQUEST and X!Tandem and computes a new *E-value* base on an *E-value* calibration procedure [1].
Default value: $-cev$ 0.
$-cev$ 0 = *E-value* calibration off.
$-cev$ 1 = *E-value* calibration on.

−cg

Chemical group attached to peptide *C-terminal*.
Default value: $-cg$ 17.002739
$-cg$ 17.002739 = Free Acid
$-cg$ 16.01872 = Amide

−daa

This option is used with RAId_aPS.
A list of the allowed amino acid residues separated by comma without space can be specified to be used by RAId_aPS.
Default value:
$-daa$ $A00, G00, V00, L00, I00, P00, F00, Y00, W00, S00, T00, C00, M00, N00, Q00, D00, E00, K00, R00, H00$.
Any of the amino acids and modifications presented in the file RAId_DbS_PTM_file are allowed choices .
The example below includes 2 PTMs G01 and G02
$-daa$ $A00, G00, G01, G02, V00, L00, I00, P00, F00, Y00, W00, S00, T00, C00, M00, N00, Q00, D00, E00, K00, R00, H00$

−db

This option is used to specified the protein database to be searched.
$-db$ /path/database_name

−dnf

This option is used with RAId_aPS.
A flat file single column containing a list of peptides to be score using *de Novo* statistics.
The $-dnf$ options also takes as input the output files from **X!Tandem (.xml), Mascot (.dat) and SEQUEST**

(.out).
Below is an example of the input file format for a list of peptides to be score.
NYQEAKDAFLGSFLYEYSR
NYKAKQGGLRFAHLLDQVSR
LLAQQSLNQQYLNHPPPVSR
KAYDLQSDAIYKADLEWLR
. . .
. . .

−dsv
This option is used with RAId_aPS.
Scoring functions to score peptides using RAId_aPS algorithm.
Any combination of the different scoring functions separated by comma are allowed parameters.
Default value: $-dsv$ 1
Allowed options
$-dsv$ 1 = RAID_DbS score.
$-dsv$ 2 = RAId(Kscore).
$-dsv$ 3 = RAId(Hyperscore).
$-dsv$ 4 = RAId(XCorr).
$-dsv$ 1,2,3 = will compute the *score, E-value* and *P-value* using the three scoring function RAID_DbS, RAId(Kscore) and RAId(Hyperscore).

−dt
Daughter fragment ions mass tolerance (Da.).
Default value: $-dt$ 0.2.

−ed
User can used the $-ed$ to specify any experimental details to the final output file.
Users must use quotation marks to specify the experimental details.
Example:
$-ed$ "Human liver cancer cell line study."

−evc
Maximum *E-value* associated with reported peptides.
Default value: $-evc$ 50.

−ex
RAId execution mode.
Default value: $-ex$ 1
$-ex$ 1 = RAId_DbS mode.
$-ex$ 2 = RAId_aPS mode
$-ex$ 3 = RAId_aPS mode reassign E-value to spectral library
$-ex$ 4 = RAId computes proportion of false discovery

−exf
Extracting MS/MS spectrum file option.
RAId can process multiple or single MS/MS spectrum file in the following formats: **.dta, .mgf, .mzXML, .pepXML, .pkl** and also spectral library files **.mgf, .msp**.
The $-exf$ option will generate single MS/MS spectrum from a file containing multiple MS/MS spectra and the extracted MS/MS files can then be used by RAId.
$-exf$ file_name

−exo
Extracting X!Tandem multiple output results combined in a single .xml file.
The $-exo$ option will generate single .xml output files that can be further processed by RAId.
$-exo$ file_name

−ez
Enzyme option.
Default value: $-ez$ 1

$-ez$ 0 = C-terminal cleavage of (A,G,V,L,I,P,F,Y,W,S,T,C,M,N,Q,D,E,K,R,H)
$-ez$ 1 = Trypsin (K,R)
$-ez$ 2 = Lys-C (K)
$-ez$ 3 = Arg-C (R)
$-ez$ 4 = GluC-Phosphate (E,D)
$-ez$ 5 = GluC-Bicarbonate (E)
$-ez$ 6 = Pepsin (L,F)
$-ez$ 7 = Chymotrypsin (F,Y,W)

−fp
Formatting database option.
The option $-fp$ will generate a database that can be used by RAId from a file of protein sequences in FASTA format.
$-fp$ -i input_database_name -o output_database_name

−gp
Searching for a peptide inside RAId's database.

./RAId -gp query_peptide -db database_name

−IL
Consider amino acids I and L the same when estimating proportion of false discovery.
Default value: $-IL$ 1. I equals to L.
$-IL$ 0 = I different from L.

−ip
Input the directory path and file name of the MS/MS spectrum file to be analyzed.
$-ip$ /path/msms_filename

−lb
Lower bound in the charge state of the precursor ion.
Default value: $-lb$ 0.
Option -lb 0 together with option -ub 0 will search spectrum using the charge state and molecular weight present in the MS/MS file.
Option $-lb$ 1 will start searching database using charge state +1 for precursor ion.
Allowed range values $[1, 9]$.

−mc
Cysteine modification options.
Default value: $-mc$ C00 Unmodified cysteine (103.009186 Da.).
Chemical group attached to the side chain of cysteine.
Other cysteine modifications can be found in the file RAId_DbS_PTM_file. If the user cysteine modification is not present in RAId_DbS_PTM_file the user can add to the file the modify cysteine value.
$-mc$ C00 = Unmodified Cysteine (103.009186 Da.).
$-mc$ C31 = Carboxymethylation (161.014649 Da.).
$-mc$ C32 = Carbamidomethylation (160.030646 Da.).
$-mc$ C33 = Pyridylethylation (208.066421 Da.).

−mw
This option is used with RAId_aPS.
Molecular weight (mw) used to compute the total number of possible peptides when a MS/MS file is not provided to RAId_aPS.
$-mw$ 2354.34 . Will compute the total number of peptides for the requested molecular weight.
The allowed molecular weight range for $-mw$ is between [57,5000].

−ng
Chemical group attached to peptide *N-terminal*.
Default value: $-ng$ 1.007825
User can specify any molecular weight after *-ng*.
Example:
$-ng$ 1.007825 = Hydrogen.

$-ng$ 43.01838 = Acetyl.

−ns
Searching for novel single amino acid polymorphisms (SAPs).
Default value: $-ns$ 0.
$-ns$ 0 = does not search for SAP.
$-ns$ 1 = does search for SAP.

−of
Specify the name for the output file.
$-of$ output_file_name

−op
Output search results path.
$-op$ /path/

−pf
The parameters used with RAId can be set by creating a parameter file.
See file RAId_parameters.txt as an example.
The parameter file is specified with the **−pf** option.
$-pf$ RAId_parameters.txt

−pfd
Option used to compute proportion of false discovery (PFD).
Default value: $-pfd$ 0. Computes proportion of false discovery using fractional count by applying Soric's formula.
$-pfd$ 1 = use to test RAId's computed E-value using a standard protein set.
$-pfd$ 2 = use to test RAId's computed PFD using a standard protein set.
$-pfd$ 3 = generates a non-redundant list of identified peptides.

−pt
Parent ion mass tolerance (Da.).
Default value: $-pt$ 1.0.
RAId will look for all masses within ± 3×(Parent ion mass tolerance).

−QK
Consider amino acids Q and K the same when estimating proportion of false discovery.
Default value: $-QK$ 0. Q different from K.
$-QK$ 1 = Q equals to K.

−rap
Users can specify the Post-Translational Modified (PTMs) residues separated by comma without space to be consider during database search.
Default value: all annotated PTMs will be consider when the -ap is different from zero.
$-rap$ $P01$ = will allowed annotated PTM of proline P01.
$-rap$ $P02, K03$ = will allowed annotated PTMs of proline (P01) and lysine (K03).

−ras
Users can specify Single Amino Acid Polymorphisms (SAPs) residues to be consider during database search.
Default value: all annotated SAPs will be consider when the $-as$ is different from zero.
Any of the 20 standard amino acids separated by comma without any space are allowed as parameter for the $-ras$ field.
$-ras$ P = will search only for annotated SAP of proline (P).
$-ras$ P, K = will search for annotated SAPs of proline and lysine.

−rtf
Contains the hydrophobicity for individual amino acid residues in a 2 column file format.
First column is the 3-letter code for the amino acid residue as provided in the file "RAId_DbS_PTM_file" followed by the amino acid hydrophobicity.
File format example:
W00 13.35

F00 11.67
L01 9.4
I02 7.96
M00 6.27
V00 4.68
Y00 5.35
... ...
... ...

−rti
y-intercept used to compute peptide retention time.
Default value: $-rti$ 0
The input value used for $-rti$ is extracted form the calibration plot of retention time versus hydrophobicity.

−rto
Option used to compute peptides hydrophobicity (retention time).
Defaul value: $-rt$ 2
Allowed options
$-rt$ 1 = User specify hydrophobicity file.
$-rt$ 2 = Uses amino acid hydrophobicity computed from a 300A pore column.
$-rt$ 3 = Uses amino acid hydrophobicity computed from a 100A pore column

−rts
Slope used to compute peptide retention time.
Defaul value: $-rts$ 1.
The input value used for $-rts$ is extracted form the linear plot of retention time versus hydrophobicity [?] .

−sm
MS/MS data collection mode.
Default value: $-sm$ 1.
$-sm$ 0 = Profile mode.
$-sm$ 1 = Centroid mode.

−spf
Users can customize their scoring function by modifying the file Score_Parameter_File located in the directory PARAMETER_FILES.
$-spf$ /Path/Score_Parameter_File
Scoring peptides using the customized scoring series in the file Score_Parameter_File.

−st
When searching for novel SAPs this option control the number of allowed single nucleotide change to occur for each of the 3-letter codon for the standard amino acids. default value: $-st$ 1.
$-st$ 1 = one nucleotide mutation per amino acid codon allowed.
$-st$ 2 = two nucleotide mutations per amino acid codon allowed.
$-st$ 3= three nucleotide mutations per amino acid codon allowed.

−tpp
Sending a peptide to be scored.
default value: $-tpp$ equal empty string.
$-tpp$ DGAGDVAFIR = the peptide DGAGDVAFIR will be scored.

−tpf
Fasta file containing true positive proteins.
$-tpf$ file_name

−ub
Upper bound in the charge state of the precursor ion.
Default value: $-ub$ 0.
Option -lb 0 together with option -ub 0 will search spectrum using the charge state and molecular weight present in the MS/MS file.

Option $-ub$ 3 will stop searching database using charge +3 state for precursor ion.
Allowed range values $[1, 9]$.

−ud
User annotated database. See the section on database formatting for more details on how to construct a user knowledge database.
$-ud$ /path/user_database_name

−up
Number of user requested Post-Translation Modifications (PTMs) allowed per peptide.
Default value: $-up$ 0.
Allowed parameter range $[0, 5]$.
$-ap$ 0 = does not search for user requested PTMs.
$-ap$ 1 = 1 user requested PTM allowed per peptide.
$-ap$ 2 = 2 user requested PTMs allowed per peptide.

−urp
Users can specify novel Post-Translational Modified (PTMs) residues not yet annotated in RAId's database.
Default value: NONE
$-urp$ $P01$ = annotated PTM of proline P01.
$-urp$ $P02, K03$ = annotated PTMs of proline (P01) and lysine (K03).

−v
Will print RAId's current version.
./RAId $-v$

E. RAId Enhanced Organism Databases Status

Organism	DB_name	Protein	NP	NM	SP	PPs	SAPs	PTMs	DB_size (byte)
Homo sapiens	hsa	29284	35059	35031	15030	271557	116073	84406	16,265,018
Anopheles gambiae	angam	12388	12719	127067	112	0	350	50	6,042,277
Arabidopsis thaliana	artha	29651	31740	31711	5527	48707	5207	11977	12,318,213
Bos taurus	botau	23796	26504	26491	3979	102130	3295	15810	11,188,490
Caenorhabditis elegans	caele	22563	23097	23097	2890	0	1045	7756	10,050,609
Canis familiaris	cafam	31705	33834	33821	528	65224	2766	4196	18,458,474
Danio rerio	darer	31192	36150	36137	1552	37113	7358	3841	14,477,794
Drosophila melanogaster	drmel	17232	20207	20207	2568	69104	5611	9290	9,796,785
Equus caballus	eqcab	17300	17637	17624	171	101657	485	1045	9,404,150
Gallus gallus	gagal	18154	18724	18681	1455	52576	1109	6522	8,728,501
Macaca mulatta	mamul	32547	38141	38128	207	139505	1370	1262	14,498,187
Mus musculus	mumus	28506	35503	35451	12170	179017	27614	61684	14,363,491
Oryctolagus cuniculus	orsat	26636	26784	26777	1205	66014	1291	2182	10,679,924
Oryza sativa	orsat	26636	26784	26777	1205	0	1291	2182	10,679,924
Pan troglodytes	patro	41464	52130	52117	482	166691	3721	3734	20,217,986
Rattus norvegicus	ranor	28914	39425	39389	5569	1181710	9297	33240	15,879,569
Saccharomyces cerevisiae	sacer	5699	5880	0	5807	95456	5507	13220	2,927,330

Table 1. The header abbreviations in this table are explained as follows. The second column, headed by DB_name, documents the abbreviated database name for searches using standalone version of RAId_DbS. The column headed by "Protein" indicates the final number of protein clusters in the processed organismal databases. The columns headed by NP, NM, and SP summarize the break down of the total number of accession numbers included respectively from protein products, transcript products, and SwissProt protein entries. The columns headed by PPs, SAPs and PTMs indicate respectively the total number of annotated proteotypic peptides (PP), single amino acid polymorphisms (SAPs) and post-translational modifications (PTMs) included. The last column shows the database size in bytes

F. Database Formatting

RAId provides users with a series of annotated databases. However, if users want to use a different database they can do so by first formatting the database. The database to be format has to be a file in **FASTA format** and the database can be easyly format by using $-fp$ option.

Example:

$-fp$ input_database_filenname output_database_filename

After the database is formatted the 3 files will be created
output_database_filename.def, **output_db_filename.seq**, **output_db_filename.prs**.

The output_format_database_name is the database that can be process by RAId using $-ud$ option.

RAId also permits users to create their own knowledge database. To generate a user specified database the user need to create two files: a **FASTA file** of sequences where the first word after the symbol > will be used as the sequence identifier (>seq_identifier) and a second file containing the user expertise knowledge related to PTMs, SAPs and diseases in a **knowledge file**.

Fasta file example:
>Id_Seq1(sequence identifier).
MLLATLLLLLLGGALAHPDRIIFPNHACEDPPAVLLEVQGTLQRPLVRDSRTSPANCTWLILGSKEQTVT
IRFQKLHLACGSERLTLRSPLQPLISLCEAPPSPLQLPGGNVTITYSYAGARAPMGQGFLLSYSQDWLMC
LQEEFQCLNHRCVSAVQRCDGVDACGDGSDEAGCSSDPFPGLTPRPVPSLPCNVTLEDFYGVFSSPGYT
. . .

Knowledge file example:
>Id_Seq1

48	SAP	R	W	deadly cancer
56	PTM	N	N08,N09,N10,N11,N12	
111	PTM	N	N08,N09,N10,N11,N12	
139	SAP	M	V	diabetes
193	SAP	N	L,I,V	
193	PTM	N	N08	
299	PTM	N	N08,N09,N10,N11,N12	
365	SAP	A	T	color blind
434	SAP	S	C,T,V,P	insulin dependent diabetes
558	SAP	R	H,P,W	

The **knowledge file** structure is explained below.
>seq_identifier
First column field is the residue position.
Second column field signifies a SAP or PTM.
Third column field is the original residue present in the sequence.
Fourth column field is either a list of possible SAPs (L,I,V) or a list of possible PTMs (N08,N09,N10,N11,N12)
Fifth column field is the disease name if any at the given position.

Once the user has created the two files as described above the user can generate a format database which RAId can process by executing UserDb.pl script.

Example:

UserDb.pl fasta_file_name knowledge_file_name output_format_database_name

The output_format_database_name is the database that can be process by RAId using $-ud$ option.

G. Post-Translation Modifications (PTMs) File

RAId_DbS_PTM_file is the file that contains information relatated to amino acid residues and their corresponding post-translational modifications it is located in the directory **PARAMETER_FILES**. The user can add any new post-translational modification to this file as long as one keeps with the same annotation structure shown below.

Line Code	Description
ID	Chemical Name of Amino Acid/PTM
AC	Residue Key
TG	Target Unmodified Amino Acid
RW	Unmodified Amino Acid Molecular Weight
MW	Modified Amino Acid Molecular Weight
PA	Location of the Modification in the Amino Acid Residue
PP	Position of the Amino Residue in the Peptide
CF	Chemical Modification to the Amino Acid Residue
MM	Monoisotopic Mass Difference MM=MW-RW
KY	Other Common Names Used to Identify the Same Molecule
LT	Other Terms Found In Literature not Necessary Correct Names

Some examples of the addition of new residues to the **RAId_DbS_PTM_file** file.

ID	Cholesterol glycine ester
AC	G01
TG	Glycine
RW	57.021465
MW	425.365766
PA	Amino acid backbone.
PP	C-terminal.
CF	C27 H44
MM	368.344301
KY	Lipoprotein.
LT	None

ID	N-palmitoyl cysteine
AC	C06
TG	Cysteine
RW	103.009186
MW	341.238852
PA	Amino acid backbone.
PP	N-terminal.
CF	C16 H30 O1
MM	238.229666
KY	Lipoprotein; Palmitate; Palmitoylation.
LT	Polmitoylation

H. Fragmentation Series Score

Each scoring function provided by RAId has its own internal default values. As an example when the RAId(XCorr) scoring function is selected by the user with the option $-dsv$=4 the default value used by the RAId(Corr) scoring function are shown in Table 2.. The first column in table are the allowed fragmentation series. In the second column in the table, the number 1 indicates that a fragmentation series is used by the scoring function and 0 indicates that a fragmentation series is not used by the scoring function. While the third column indicates the molecular weight difference from the select series relative to the b or y series, e.g. each fragment from the a_1 series differ from its corresponding b_1 fragment by exactly -27.994914 Da. The forth column indicates with a 1 if the fragmentation series belongs to the N-terminal (b) and with a 2 if the fragmentation series belongs to the C-terminal (y). The fifth column provides the weight associated with the fragment series, i.e. during scoring the evidence associated with the a series will be multiplied by a weight factor of 0.2. The last column is the charge associated with the selected series. Similar default values exist for other scoring functions used by RAId. Therefore the scoring function default values will be used when a scoring function is selected.

Users also have the option to customize the available scoring functions by modifying the values present in the file Parameter_file and passing this information to RAId with the $-spf$ option. As long as users preserve the file structure they can add new fragmentation series evidence to this file, modify series weight, include higher charge state, basically they can add or modify any field in the file Score_Parameter_file.

Series	SU	MDR(b,y)	b/y	SW	Z
b	1	0	1	1.0	1
$b-H$	1	-1.007825	1	0.5	1
$b+H$	1	1.007825	1	0.5	1
a	1	-27.994914	1	0.2	1
c	0	17.026549	1	0.0	1
$b-NH3$	1	-17.026549	1	0.2	1
$b-H2O$	1	-18.010564	1	0.2	1
b	0	0	1	0.0	2
y	1	0	2	1.0	1
$y-H$	1	-1.007825	2	0.5	1
$y+H$	1	1.007825	2	0.5	1
x	0	25.979264	2	0.0	1
$y-NH3$	1	-17.026549	2	0.2	1
$y-H2O$	0	-18.010564	2	1.0	1
y	0	0	2	0.0	2

Table 2. The table is an example of the default values used by RAId(XCorr) scoring function. Series used (SU) 1=yes,0=no. Mass difference relative (MDR) to b's or y's peaks. Series weight (SW) used by the scoring function. Series charge (Z) state.

I. RAId_DbS and RAId_aPS Command Line Examples

The most straight foward way to use RAId from the command line is by executing RAId with the following command.

(A): When executing single files

>./RAId $-pf$ RAId_parameters.txt $-ip$ /path/msms_filename

The parameters used by RAId are taking from the file RAId_parameters.txt ($-pf$ RAId_parameters.txt)

The Input MS/MS spectrum file path location $-ip$ /path/msms_filename

(B): When the MS/MS spectra are all contained a single file as in the case of files with extension PKL, mzXML, mzML, XML and mzData

>./RAId -b run_jobs $-pf$ RAId_parameters.txt $-ip$ /path/msms_filename.(PKL, mzXML, mzML, XML, mzData)

Name of bash file to be created for execution $-b$ (run_jobs).

The parameters used by RAId are taking from the file RAId_parameters.txt ($-pf$ RAId_parameters.txt)

The Input MS/MS spectrum file path location $-ip$ /path/msms_filename.(PKL, mzXML, mzML, XML, mzData)

Example 1: Generating the bash file to be executed.

Command line:

>./RAId $-ex$ 2 $-b$ run_jobs $-ez$ 1 $-dsv$ 1,2,3,4 $-dt$ 0.05 $-pt$ 0.8 $-ng$ 1.007800 $-cg$ 17.002700 $-up$ 2 $-urp$ S06,Y10 $-db$ /path/database_name $-ip$ /data_path/data_name $-op$ /output_path/

aPS executing mode $-ex$ 2.

Name of bash file to be created for execution $-b$ (run_jobs).

Trypsin as the enzyme $-ez$ 1.

Scoring functions to be used $-dsv$ 1,2,3,4.

Molecular error tolerance of daughter ion 0.05 Da. $-dt$ 0.05.

Molecular error tolerance of parent ion 0.8 Da. $-pt$ 0.05.

N-terminal group hydrogen $-ng$ 1.0078.

C-terminal group free acid $-cg$ 17.0027.

Number of user requested PTMs allowed per peptide $-up$ 2.

User requested PTMs phosphorylation of serine and tryptophan $-urp$ S06,Y10

Protein database path location $-db$ /path/database_name.

Input file name to be processed $-ip$ /data_path/data_name.

Search result output path location $-op$ /output_path/.

Example 2: Computing the total number of possible peptides within a given molecular weight.

Command line:

>./RAId $-ex$ 2 $-ez$ 1 $-mw$ 1345.67 $-dt$ 0.05 $-pt$ 0.8 $-ng$ 1.007800 $-cg$ 17.002700 $-op$ /path/

aPS executing mode $-ex$ 2.

Trypsin as the enzyme $-ez$ 1.

Requested molecular weight $-mw$ 1345.67

Molecular error tolerance of daughter ion 0.05 Da. $-dt$ 0.05.

Molecular error tolerance of parent ion 0.8 Da. $-pt$ 0.05.

N-terminal group hydrogen $-ng$ 1.0078.

C-terminal group free acid $-cg$ 17.0027.

Search result output path location $-op$ /output_path/.

Example 3: Generating the score distribution for all possible peptides.

Command line:

>./RAId $-ex$ 2 $-ez$ 1 $-daa$ A00,G00,G02,G03,V00,L00,F00,Y00,W00,S00,T00,C00,N00,Q00,D00,E00,R00 $-dt$ 0.05 $-pt$ 0.8 $-ng$ 1.007825 $-cg$ 17.002739 $-dsv$ 4 $-ip$ /path/msms_filename $-op$ /path/

aPS executing mode $-ex$ 2.

Trypsin as the enzyme $-ez$ 1.

Amino acids residues selected:
$-daa$ A00,G00,G02,G03,V00,L00,F00,Y00,W00,S00,T00,C00,N00,Q00,D00,E00,R00.

Molecular error tolerance of daughter ion 0.05 Da. $-dt$ 0.05.

Molecular error tolerance of parent ion 0.8 Da. $-pt$ 0.05.

N-terminal group hydrogen $-ng$ 1.0078.

C-terminal group free acid $-cg$ 17.0027.

Scoring function XCorr selected to compute score histogram $-dsv$ 4.

Input MS/MS spectrum file path location $-ip$ /path/msms_filename.

Search result output path location $-op$ /path/.

Example 4: Using RAId_aPS to re-score the output files from SEQUEST, X!Tandem or Mascot.

Command line:

>./RAId $-ex$ 2 $-dsv$ 1,2,3,4 $-mc$ C32 $-ng$ 1.007825 $-cg$ 17.002739 $-pt$ 1.0 $-dt$ 0.2 $-dnf$ /path/sequest_output_file.out $-ip$ /path/msms_filename $-op$ /path/

aPS executing mode $-ex$ 2.

Scoring functions RAId_DbS, Hyperscore, XCorr and Kscore selected $-dsv$ 1,2,3,4.

Cysteine modification $-mc$ C32.

N-terminal group hydrogen $-ng$ 1.0078.

C-terminal group free acid $-cg$ 17.0027.

Molecular error tolerance of parent ion 1.0 Da. $-pt$ 1.0.

Molecular error tolerance of daughter ion 0.2 Da. $-dt$ 0.2.

SEQUEST, X!Tandem or Mascot output files $-dnf$ /path/sequest_output_file.out.

Input MS/MS spectrum file path location $-ip$ /path/msms_filename.

Search result output path location $-op$ /path/.

Example 5: Using RAId_aPS to rescore the output files from SEQUEST, X!Tandem or Mascot and also computing an *E-value* based on the original score.

Command line:

>./RAId $-ex$ 2 $-dsv$ 1,2,3,4 $-mc$ C32 $-ng$ 1.007825 $-cg$ 17.002739 $-pt$ 1.0 $-dt$ 0.2 $-dnf$ /path/sequest_output_file.out $-sm$ 1 $-cev$ 1 $-ip$ /path/msms_filename $-op$ /path/

aPS executing mode $-ex$ 2.

Scoring functions RAId_DbS, Hyperscore, XCorr and Kscore selected $-dsv$ 1,2,3,4.

Cysteine modification $-mc$ C32.

N-terminal group hydrogen $-ng$ 1.0078.

C-terminal group free acid $-cg$ 17.0027.

Molecular error tolerance of parent ion 1.0 Da. $-pt$ 1.0.

Molecular error tolerance of daughter ion 0.2 Da. $-dt$ 0.2.

SEQUEST, X!Tandem or Mascot output files $-dnf$ /path/sequest_output_file.out.

MS/MS spectrum acquisition mode centroid $-sm$ 1.

Computes *E-value/P-value* using peptides original score $-cev$ 1.

Input MS/MS spectrum file path location $-ip$ /path/msms_filename.

Search result output path location $-op$ /path/.

Example 6: Using RAId_aPS as a database search tool.

Command line:

>./RAId $-ex$ 2 $-ez$ 1 $-dsv$ 1,3 $-mc$ C32 $-ng$ 1.007825 $-cg$ 17.002739 $-pt$ 1 $-dt$ 0.2 $-db$ /path/database_name $-ip$ /path/msms_filename $-op$ /path/

aPS executing mode $-ex$ 2.

Trypsin as the enzyme $-ez$ 1.

Scoring functions RAId_DbS and Hyperscore selected $-dsv$ 1,3.

Cysteine modification $-mc$ C32.

N-terminal group hydrogen $-ng$ 1.0078.

C-terminal group free acid $-cg$ 17.0027.

Molecular error tolerance of parent ion 1.0 Da. $-pt$ 1.0.

Molecular error tolerance of daughter ion 0.2 Da. $-dt$ 0.2.

Protein database path location $-db$ /path/database_name.

Input MS/MS spectrum file path location $-ip$ /path/msms_filename.

Search result output path location $-op$ /path/.

J. RAId_DbS and RAId_aPS Output File Fields Description

Operation mode	RAId_DbS or RAId_aPS
Date	Date code was executed
Experimental description	Any details related to experimment
Input file	MS/MS input file name
Output file	Output file name
Data acquisition mode (Profile,Centroid)	Data acquisition mode
MS/MS scan number	Scan number of current MS/MS
MS/MS scan total ion current	Total ion current present in the MS/MS
MS/MS scan precursor ion intensity	Ion current of the precursor ion
Observed molecular weight	Experimental computed molecular weight in charge +1 state
Observed charge state	Experimental observed charge state
Lowest charge state searched	Lowest charge assume for precursor ion
Highest charge state searched	Highest charge state assume for precursor ion
Parent ion molecular weight tolerance	Error associated with precursor ion
Daughter ion molecular weight tolerance	Error associated with fragment ions
Retention time	Experimental observed peptide elution time
Database searched	Name of database searched
User knowledge database searched	Name of specialized user knowledge database serched
Total database size (bytes)	Number of bytes in the database
Total number of unique peptides	Total number of unique peptides score
Proteptypic peptide information source	Spectral library key:Spectral library file name
Enzyme cleavage site	Enzyme cleavage site
Number of allowed PTMs per peptide	Allowed number of PTMs per peptide
Number of allowed SAPs per peptide	Allowed number of SAPs per peptide
Searching for novel SAPs	Searching for novel SAPs
Requested database annotated SAPs	Allowed annotated SAPs
Requested database annotated PTMs	Allowed annotated PTMs
Cysteine modification	Cysteine modification
Cysteine molecular weight	Cysteine modification molecular weight
Cysteine modification name	Cysteine modification name
N-terminal modification	Group present in the N-terminal
C-terminal modification	Group present in the C-terminal
Goodness-of-Fit Model	Agreement between experimental and theoretical distributions
Model Correctness P-value	The probability that the agreement is a random match
Degrees of Freedom Fit	Model degrees of freedom
Average weighted peak counts user per peptide	Weighted average number of evidence used per peptide
Mean of score (S) distribution	$\bar{S} = \sum_{i=1}^{N} S_i/N$
Variance of score distribution	$\sum_{i=1}^{N} (S_i - \bar{S})^2/N$
Skewness of score distribution	$\sum_{i=1}^{N} (S_i - \bar{S})^3/N$
Maximum E-value reported	Peptide E-value reported cutoff
Total number of possible peptides	Total number of peptides score during RAId_aPS
Total number of mass grid used	Number of mass grid used to generate RAId_aPS Peptides
Maximum peptide length	Maximum peptide length for input precursor ion
Minimum peptide Length	Minimum peptide length for input precursor ion
Rank	Rank of the scored peptide by E-value
Peptide	The peptide sequence
Pre residue	The N-terminal residue prior to the peptide sequence
Post residue	The C-terminal residue prior to the peptide sequence
Postranslation modification position	Position PTM in the peptide sequence

Postranslation modification	RAId_DbS 3-letter code for the PTM
Postranslation modification molecular weight	PTM molecular weight
Postranslation modification name	PTM common name
Protein identification code	Protein associated with peptide identification code
Peptide protein position	Position of peptide in the protein
Peptide molecular weight	Theoretical peptide molecular weight
Peptide delta mass	Experimental minus theoretical molecular weight
Peptide retention time (hydrophobicity)	Peptide theoretical computed hydrophobicity
Peptide proteotypic information	Spectral library key(1,2,3,4,...):consensus spectrum index,
Novel single amino acid polymorphism	Location of novel SAP in the peptide
Annotated literature disease information	Literature information related to diseases
RAId_DbS score	Score computed using RAId_DbS scoring function
RAId_DbS E-value	E-value computed using RAId_DbS scoring function
RAId_DbS P-value	P-value computed using RAId_DbS scoring function
RAId_aPS Hyperscore score	Score computed using RAId_DbS Hyperscore function
RAId_aPS Hyperscore P-value	E-value computed using RAId_DbS Hyperscore function
RAId_aPS Hyperscore E-value	P-value computed using RAId_DbS Hyperscore function
RAId_aPS XCorr score	Score computed using RAId_DbS XCorr function
RAId_aPS XCorr P-value	E-value computed using RAId_DbS XCorr function
RAId_aPS XCorr E-value	P-value computed using RAId_DbS XCorr function
RAId_aPS Kscore score	Score computed using RAId_DbS Kscore function
RAId_aPS Kscore P-value	E-value computed using RAId_DbS Kscore function
RAId_aPS Kscore E-value	P-value computed using RAId_DbS Kscore function

Table 3.: RAId_DbS output file fields description.

K. RAId_DbS Output Example

RAId output begins
Operation mode = RAId_DbS
Date = Mon Apr 6 14:40:32 2009
Experimental description = Bovine protein example
Input file = BOVINE19.dta
Output file = BOVINE19.dta.Mon_Apr_6_14-40-32_2009
Data acquisition mode (Profile,Centroid) = Profile
Observed molecular weight = 2301.080078
Observed charge state = 3
Lowest charge state searched = 3
Highest charge state searched = 3
Parent ion molecular weight tolerance = 1.000000
Daughter ion molecular weight tolerance = 0.200000
Database searched = /DATABASE/botau
User knowledge database searched = None
Total database size (bytes) = 11328865.00
Total number of unique peptides = 9961.00
Enzyme cleavage site = RK
Number of allowed PTMs per peptide = 0
Number of allowed SAPs per peptide = 1
Searching for novel SAPs = No
Requested database annotated SAPs = D,S
Requested database annotated PTMs = S01,S02
Cysteine modification = C32
Cysteine molecular weight = 160.029996
Cysteine modification name = Carbamidomethylation
N-terminal modification = 1.007825
C-terminal modification = 17.002739
Goodness-of-Fit Model = 0.95
Model Correctness P-value = 3.4e-13
Degrees of Freedom Fit = 9.0
Average weighted peak counts user per peptide = 9.90
Maximum E-value reported = 10.000000
//
Rank = 1
Peptide = NYQEAKDAFLGSFLYEYSR
Pre residue = K
Post residue = R
Peptide protein position = 340
Protein identification code = 30794280
Peptide molecular weight = 2301.060059
Peptide retention time (hydrophobicity) = 37.91
Novel single amino acid polymorphism = None
Annotated literature disease information = None
RAId_DbS score = 1.3
RAId_DbS E-value = 2.4e-10
RAId_DbS P-value = 1.1e-13
//
Rank = 2
Peptide = LLAQQSLNQQYLNHPPPVSR
Pre residue = R
Post residue = S
Peptide protein position = 335
Protein identification code = 76608636
Peptide molecular weight = 2303.199951
Peptide retention time (hydrophobicity) = 33.50
Novel single amino acid polymorphism = None
Annotated literature disease information = None

RAId_DbS score = 0.65
RAId_DbS E-value = 0.057
RAId_DbS P-value = 9e-05
//
RAId output ends

L. RAId_aPS Output Example

RAId output begins
Operation mode = RAId_aPS
Date = Mon Apr 6 14:01:11 2009
Experimental description = Bovine protein example
Input file = BOVINE19.dta
Output file = BOVINE19.dta.aPS.4595.91.Mon_Apr_6_14-01-11_2009
Data acquisition mode (Profile,Centroid) = Profile
Observed molecular weight = 2301.080078
Observed charge state = 3
Lowest charge state searched = 3
Highest charge state searched = 3
Parent ion molecular weight tolerance = 1.000000
Daughter ion molecular weight tolerance = 0.200000
Database searched = /DATABASE/botau
User knowledge database searched = None
Total database size (bytes) = 11328865.00
Total number of unique peptides = 9961.00
Enzyme cleavage site = RK
Number of allowed PTMs per peptide = 0
Number of allowed SAPs per peptide = 1
Searching for novel SAPs = No
Requested database annotated SAPs = D,S
Requested database annotated PTMs = S01,S02
Cysteine modification = C32
Cysteine molecular weight = 160.029996
Cysteine modification name = Carbamidomethylation
N-terminal modification = 1.007825
C-terminal modification = 17.002739
Total number of possible peptides = 2.99e+26
Total number of mass grid used = 2036
Maximum peptide length = 39
Minimum peptide Length = 13
Maximum E-value reported = 10.00
//
Rank = 1
Peptide = NYQEAKDAFLGSFLYEYSR
Pre residue = K
Post residue = R
Peptide protein position = 340
Protein identification code = 30794280
Peptide molecular weight = 2301.060059
Peptide retention time (hydrophobicity) = 37.91
Novel single amino acid polymorphism = None
Annotated literature disease information = None
Combined database E-value = 2.5e-35
RAId_aPS RAId score = 1.28
RAId_aPS RAId E-value = 1.16e-14
RAId_aPS RAId P-value = 5.42e-18
RAId_aPS Hyperscore score = 44.16
RAId_aPS Hyperscore E-value = 7.14e-06
RAId_aPS Hyperscore P-value = 3.34e-09
RAId_aPS XCorr score = 4.05
RAId_aPS XCorr E-value = 6.11e-08
RAId_aPS XCorr P-value = 2.86e-11
RAId_aPS Kscore score = 248.00
RAId_aPS Kscore E-value = 1.02e-09
RAId_aPS Kscore P-value = 4.77e-13
//

Rank = 2
Peptide = PVPSGPAGGSGGGGSGGGGGGQPPPLQR
Pre residue = R
Post residue = G
Peptide protein position = 1029
Protein identification code = 119915148
Peptide molecular weight = 2299.117995
Peptide retention time (hydrophobicity) = 25.28
Novel single amino acid polymorphism = None
Annotated literature disease information = None
Combined database E-value = 5.1
RAId_aPS RAId score = 0.59
RAId_aPS RAId E-value = 5.22e+00
RAId_aPS RAId P-value = 8.55e-03
RAId_aPS Hyperscore score = 22.81
RAId_aPS Hyperscore E-value = 3.28e-01
RAId_aPS Hyperscore P-value = 5.37e-04
RAId_aPS XCorr score = 0.45
RAId_aPS XCorr E-value = 2.06e+02
RAId_aPS XCorr P-value = 3.37e-01
RAId_aPS Kscore score = 92.00
RAId_aPS Kscore E-value = 3.74e+01
RAId_aPS Kscore P-value = 6.13e-02
//
RAId output ends

M. Computing Proportion of False Discovery using RAId

Example 1: Using RAId to compute proportion of false discovery (PFD) using fractional count in Soric's formula.

Command line:

>./RAId $-ex$ 4 $-pfd$ 0 $-evc$ 10 $-ip$ path_containing_RAId_output $-of$ name_of_output_file

Executing mode $-ex$ 4.

Computes PFD using Soric's formula $-pfd$ 0.

Include peptides with E-value less than 10 $-evc$ 10.

Directory containing RAId results $-ip$ path_containing_RAId_output.

Name of file to write computed PFD $-of$ name_of_output_file.

Example 2: Using RAId to generate a non-redundant list of identified peptides.

Command line:

>./RAId $-ex$ 4 $-pfd$ 3 $-evc$ 1 $-ip$ path_containing_RAId_output $-of$ name_of_output_file

Executing mode $-ex$ 4.

Generating non-redundant list of identified peptides $-pfd$ 3.

Include peptides with E-value less than 1 $-evc$ 1.

Directory containing RAId results $-ip$ path_containing_RAId_output.

Name of file to write computed PFD $-of$ name_of_output_file.

References

Alves, G., A. Y. Ogurtsov, W. W. Wu, G. Wang, R. F. Shen, and Y. K. Yu, 2007, Biol. Direct **2**, 26.
Alves, G., A. Y. Ogurtsov, and Y. K. Yu, 2007, Biol. Direct **2**, 25.
Alves, G., A. Y. Ogurtsov, and Y. K. Yu, 2008, BMC Genomics **9**, 505.
Alves, G., W. W. Wu, G. Wang, R. F. Shen, and Y. K. Yu, 2008, J. Proteome Res. **7**, 3102.
Alves, G., and Y. K. Yu, 2008, Physica A **387**, 6538.
Eng, J. K., A. L. McCormack, and J. R. Yates III, 1994, J. Amer. Soc. Mass Spectrom. **5**, 976.
Feny, D., and R. C. Beavis, 2003, Anal. Chem. **75**, 768.
Keller, A., J. Eng, N. Zhang, X. J. Li, and R. Aebersold, 2005, Mol. Syst. Biol. **1**, 2005.0017.